2-5-2014

Randomized and Efficient Authentication in Mobile Environments

Wei Jiang
Missouri University of Science and Technology, wjiang@mst.edu

Dan Lin
Missouri University of Science and Technology, lindan@mst.edu

Feng Li
Missouri University of Science and Technology, lftrd@mst.edu

Elisa Bertino
Purdue University, bertino@cs.purdue.edu

Randomized and Efficient Authentication in Mobile Environments

Wei Jiang [1] Dan Lin [2] Feng Li [3] Elisa Bertino [4]

[1,2,3]Dept. of Computer Science, Missouri S & T
500 W. 15th St., Rolla, MO 65409
[4]Dept. of Computer Science, Purdue University
305 N. University Street, West Lafayette, IN 47907

[1,2,3]{wjiang, lindan, lftrd}@mst.edu
[4]bertino@cs.purdue.edu

Abstract

In a mobile environment, a number of users act as a network nodes and communicate with one another to acquire location based information and services. This emerging paradigm has opened up new business opportunities and enables numerous applications such as road safety enhancement, service recommendations and mobile entertainment. A fundamental issue that impacts the success of these applications is the security and privacy concerns raised regarding the mobile users. In that, a malicious user or service provider can track the locations of a user traveled so that other malicious act can be carried out more effectively against the user. Therefore, the challenge becomes how to authenticate mobile users while preserving their actual identity and location privacy. In this work, we propose a novel randomized or privacy-preserving authentication protocol based on homomorphic encryption. The protocol allows individual users to self generate any number of authenticated identities to achieve full anonymity in mobile environment. The proposed protocol prevents users being tracked by any single party including peer users, service providers, authentication servers, and other infrastructure. Meanwhile, our protocol also provides traceability in case of any dispute. We have conducted experimental study which demonstrates the efficiency of our protocol. Another advantage of the proposed protocol is lightweight computation and storage requirement, particularly suitable for any mobile devices with limited computation power and storage space.

1 Introduction

Due to a wide spread use of mobile devices, a user can access various location based services or communicate with peer users (within certain proximity) almost everywhere he or she goes, through dynamically and temporarily formed networks such as mobile ad-hoc network (MANET) and vehicular ad-hoc network (VANET). Considering the large number of mobile users, these ad-hoc networks open up tremendous business opportunities, and numerous mobile applications have been proposed ranging from location-based recommendation services, driving safety enhancement [7, 14], dynamic route planning [24] to mobile entertainment [39]. For example, a user may send inquiries to other users around certain landmarks to obtain the up-to-date congestion information, the condition of a road or parking information. Users can exchange files or chat with others in a newly established social network.

One of the key component towards the successful roll-out of mobile or location based applications is to provide security and privacy guarantees. Without proper security and privacy guarantees, the rich functionality and services provided by mobile ad-hoc networks can be abused, jeopardizing the safety of users, as well as the performance of the entire network. For example, a malicious user can claim a fake traffic jam to gain the right of the road and cause other vehicles to make an unnecessary detour. The user can also send

unfounded negative comments regarding a local business to other users within the same mobile network. As a result, users should be authenticated before they are allowed to access services offered through these dynamically formed mobile networks.

Since a user's location can reveal the actual identity of the user, it is in the best interest of the user not to be tracked by service providers or peer users. In many non critical scenarios, a service provider may only need to know whether the user is authenticated or not, but does not need to know the user's actual identity. Thus, users' privacy should be preserved during authentication in that their identities should be kept private in order to avoid unlawful tracing and user profiling. More specifically, these parties may cause privacy concerns: (i) the authentication server or service provider; (ii) peer users. The server may obtain the behavior pattern or track the user locations according to the record of the users requesting for authentication. Similarly, other peer users may also be able to track one another through the authentication records. We refer to the privacy concern caused by the server as the server-wise privacy and the privacy concern caused by peer users as the peer-wise privacy. Ideally, we should preserve both server-wise and peer-wise privacy for each mobile user. On the other hand, we should also ensure traceability whereby law enforcement authorities can reveal a user's real identity required when disputes occur.

Efforts have been made on developing privacy-preserving authentication protocols in both MANET and VANET. These existing protocols typically employ one of the following two strategies. One strategy is to equip users with a large number of authenticated pseudonyms [26, 30]. Then, users use authenticated pseudonyms to communicate in these ad-hoc networks so that their real identities are hidden from peer users. In most of such approaches, there are two major limitations. First, the server which produces the pseudonyms can track the users. Second, the revocation of the long list of pseudonyms of a malicious user is very costly. The other existing common strategy is to hide a user in a group of users (like in VANET [23,31,42]). Under this type of protocols, users can prove that they are valid group members without revealing real identities to other users in the same group. However, the users can still be tracked by their group manager. It is also not reasonable to assume a group is trustworthy without providing any evidence. How to establish such trust is still an open problem.

In [3, 4], an anonymous credential system that use zero-knowledge proof was proposed to achieve anonymous authentication. Being the best among the exiting work, this scheme achieves several of our proposed goals under privacy-preserving user authentication, but comparing to the proposed solution, the scheme is not very efficient and it is only statistically secure. We will adopt this scheme as a baseline to demonstrate the advantages of our proposed protocol.

1.1 Our Contribution

To overcome the shortcomings in existing work, we propose a novel randomized / privacy-preserving authentication protocol (namely RAU) that truly preserves users' privacy while still ensure traceability. The proposed protocol is designed based on Homomorphic encryption [25] and it allows each user to self generate any number of authenticated identities to prove his or her legal status when communicating with peer users, service providers, or other infrastructure (like road-side units in VANET). In fact, users will be able to easily use a new identity for each newly established communication. These randomized identities can be verified through the collaboration of a pair of authentication servers while each authentication server would not know the real identity of the authentication requester. In this way, we achieve both peer-wise and server-wise privacy preservation. For traceability, the pair of authentication servers need to execute a collaborative protocol so that the real identity of the malicious user can be identified. We summarize the advantages of our proposed authentication protocol as follows.

- Under our authentication protocol, users' real identities are hidden from each individual party including authentication servers, peer users, service providers, and other infrastructure.
- Our protocol achieves a set of desired security and privacy properties such as unforgeability, unlinkability and traceability. It is robust against various types of attacks (as discussed later in Section 5).

- Our approach no longer has the key revocation problem neither the costly group management. Specifically, users using the proposed protocol no longer need to preload a huge number of keys (i.e., pseudonyms) or rely on others (i.e., peers or infrastructure) to generate the pseudonyms. Our experimental study demonstrates the proposed protocol is very efficiency.
- Our protocol does not require users to be equipped with high performance computing equipment since almost all computations are outsourced to the servers and the users only need to generate several encryptions and random numbers.
- User authentication is very efficient in our protocol well under the 100ms requirement [20]. Since anonymity revocation needs not to be done as a real-time application (due to court orders), our protocol provides reasonable computation time (as presented in Section 6).

The rest of the paper is organized as follows. Section 2 reviews related works. Section 3 introduces some preliminary notions of encryption adopted in this work. Section 4 presents the proposed randomized authentication protocol. Section 5 discusses the possible attacks and the security and privacy properties of the protocol. Then, Section 6 reports the experimental results. Finally, Section 7 concludes the paper and outlines future research directions.

2 Related Work

Dynamically and temporarily formed mobile network can classified into two common categories: mobile ad-hoc network (MANET) and vehicular ad-hoc network (VANET). We can consider VANET as a domain specific example of MANET. In this paper, we do not distinguish between the two types of mobile networks since user privacy issues are common for both networks and our proposed authentication protocol works for cases. On the other hand, to present the related work, we separate the two networks.

2.1 Privacy-Preserving Authentications in VANET

We can categorize the existing work on privacy-preserving authentication in VANET into two major categories: (i) pseudonym-based and (ii) group-based protocols.

The general goal of the pseudonym-based authentication protocols is to enable vehicles to use different pseudonyms during communication rather than using their real identities. One of the earliest work in this category is by Raya and Hubaux [26]. They suggested that when a vehicle needs to sign a message, it randomly selects a private key from a huge pool of certificates issued by the authority. The message receiver will verify the sender's signature by checking the validity of the corresponding public key certificate. The problem of this protocol is that vehicles need to check a long list of revoked certificates when verifying each received signed-message, which is very time consuming.

Raya, et al. in [27] proposed efficient revocation schemes. However, these schemes violate the location privacy requirement and are subject to a movement tracking attack [34]. In order to reduce the average overhead of message authentication, Calandriello et al. [2] proposed a hybrid scheme, which is also computationally expensive because it needs to check if the group signature is from a revoked vehicle [35]. Other pseudonym-based protocols can be found in [13,30,32,33,41–43], achieving different degrees of improvement over the key revocation problem. However, in most these protocols, the identity management authority is required to maintain the certificates associated with each vehicle so as to retrieve the vehicles' real identities when disputes occurs. This allows the authority to track the vehicles' movement; hence, the vehicles' privacy is not fully preserved.

Another category of privacy preserving authentication protocols is group-based [6,12,16,19,22,23,29,31, 36,37,40,42]. The typical idea is to utilize group managers to group and authenticate vehicles, which enables vehicles to anonymously communicate with group members. In general, existing group-based protocols may have certain disadvantages: First, the group manager has all the knowledge about group members and

hence is able to track them. In our protocol, this becomes harder since the servers in our scheme needs to collude. Secondly, group managers are difficult to select because they serve as trusted parties. There are no theoretical results or comprehensive empirical studies on how to select a trustworthy group leader in VANET. In addition, Groups are also dynamic, and group managers can leave the group at any moment. New group leaders will know the private information within the group. The more dynamic the group becomes, the more private information can be leaked from the group. Therefore, we do not adopt grouped based approach in this paper.

2.2 Privacy-Preserving Authentications in MANET

Most related work in MANET is associated with authenticating the messages exchanged in the network without disclosing the actual identities of the source and the destination. Ciszkowski and Koutulski [9] provided an ANAP protocol which identifies the destination using the hash value of a user's pseudonymous. However, the problem of this scheme is how the source of a transmission can get the pseudonymous of the destination node. By assuming that such pseudonymous are public, attackers can pre-compute a table containing pairs of pseudonymous-hashes. In this way, when a packet is captured in the network, a destination node can be immediately discovered. On the other hand, when the pseudonymous is secret, then using hashes does not provide enough strong security [17].

Chou et al. [8] proposed an efficient anonymous communication protocol for peer-to-peer applications over MANETs which uses broadcast-based scheme and probability flooding control to establish multiple anonymous paths within a single query phase. The scheme uses controlled and probabilistic broadcasting to provide anonymity while avoids using step-by-step encryption/decryption and achieves lower computational complexity; however, this approach does not work for privacy-preserving user authentication.

Freudiger et al. [11] pointed out a self-organized anonymous message authentication protocol that a user can use a group of identities including his own to generate a ring signature and the successful verification reveals the only fact that the signature is generated by one of the group identities to authenticate the messages sent from a particular group. Similarly, Ren and Harn [28] proposed a (t, n)-threshold ring signature scheme to achieve anonymous authentication for communication by verifying the ring signature. As other privacy-preserving message authentication protocols, the above schemes cannot be used to anonymously authenticate the identity of a user.

Tsai et al. [38] proposes a secure anonymous authentication protocol to achieve user unlinkability under mobile wireless environment. In the authentication phase, after receiving the credential (certificate) from user, FA (Foreign Agent) forwards the credential with his signature to HA (Home Agent) who issued credentials to users in the initial phase. HA will check the validity of the credential by searching the mapping table and send back the acknowledgement message when the credential is correct. However, in such scheme, the HA can learn useful information to track a user when the FA sends authenticating information to him.

Kotzanikolaou et al. [21] presented an efficient anonymous authentication scheme that provides untraceability and unlinkability of mobile devices while accessing location-based services. The scheme using standard primitives such as zero-knowledge proofs, MACs and challenge/response. However, there are a couple of drawbacks: First, when a user U generates n different credentials, if each user possesses the same n, which would cause information leakage. If each user possesses different n, they could be tracked by an issuer or SP (Service Provider) in the verification process. Secondly, If more service providers join the network, not only more storage space and secret keys shared between the issuer and each service provider are needed, but also the number of communication messages increase exponentially. Thus, the scheme is not practical.

Camenisch et al. [3,4] introduced an anonymous credential system using zero-knowledge proof to achieve anonymous authentication. This scheme guarantees user anonymity but has a credential sharing problem where dishonest users can share their credentials with others. To solve such problem, Cesena et al. [5] stated a solution based on a hardware security module to prevent credential sharing. The security guarantee and functionality of Camenisch scheme match our requirement, so this scheme will be used as the baseline to evaluate the advantages of our proposed randomized authentication (RAU) protocol. Although the RAU

protocol has the same credential sharing problem, Cesena's work can be directly applied in our scheme. As a result, solving the credential sharing problem is not the focal point of our work.

There are other content-based authentication schemes utilize attribute encryption. For instance, Baden et al. [1] developed Persona which achieves privacy by encrypting private content and prevents misuse of a user's data through authentication under online social network. However, when a user authenticates another user or a group, the users and group need to belong to a certain category, such as "family", "friend", and the group size needs to be almost fixed. Attribute encryption is very expensive when there is a large number of users. In a dynamic mobile network such as MANET and VANET, both information and users constantly change, it is impractical or even impossible to apply attribute encryption in this problem domain. In addition, messages associated with location based services like providing traffic flow information are generally not confidential, so to save computation and storage costs, we should not encrypt these information under most situations.

3 Preliminary

For better understanding, we present a brief review of the cryptographic notions that are relevant to the construction of our authentication protocol.

3.1 Homomorphic Encryption

An additive homomorphic probabilistic public key encryption (HEnc^+) system is used as the building block in the proposed authentication protocol. Let E_{pk} and D_{sk} be the encryption and decryption functions in an HEnc^+ system with public key pk and secret key sk. Without sk, no one can discover x from $E_{pk}(x)$ in polynomial time. When the context is clear, we will omit pk and sk from the notations of the encryption and decryption functions. The HEnc^+ system has the following properties:

- The encryption function is additive homomorphic in that the product of the encryptions of x_1 and x_2 produces the encryption of $x_1 + x_2$.

$$E(x_1) * E(x_2) = E(x_1 + x_2) \tag{1}$$

- Given a constant c and $E(x)$:

$$E(x)^c = E(c * x) \tag{2}$$

- The encryption function has semantic security as defined in [15], i.e., a set of ciphertexts do not provide additional information about the plain-text to an adversary. E.g., suppose that y_1 and y_2 are the ciphertexts generated by performing the encryptions of x at different times using the same key, there is very high probability that $y_1 \neq y_2$, but $D(y_1) = D(y_2)$ holds.

Any HEnc^+ system is applicable, but in this paper, we adopt Paillier's public-key homomorphic encryption system [25] for the actual implementation due to its efficiency, particularly when the plain-text values are small. In a simplified version of Paillier, the public key is $N = p * q$, where p and q are large primes with similar size, and they are private information. In general, the size of N should be at least 1,024 bits. The encryption function is defined as follows for x:

$$E(x, r) = (N + 1)^x * r^N \mod N^2$$

where r is randomly chosen from $\mathbb{Z}^*_{N^2}$. Note that the encryption function is only based on the public key, and the group $\mathbb{Z}^*_{N^2}$ contains the elements from $\mathbb{Z}_{N^2} = \{0, 1, 2, \ldots, N^2 - 1\}$ which are co-prime to N^2. Since r is randomly selected each time a value is encrypted, $E(x, r_1) \neq E(x, r_2)$ if $r_1 \neq r_2$. On the other hand, $D(E(x, r_1)) = D(E(x, r_2)) = x$ regardless the value of r_1 and r_2.

In this paper, we will also use Diffie-Hellman key exchange protocol [10] to generate a random number shared by two parties or entities.

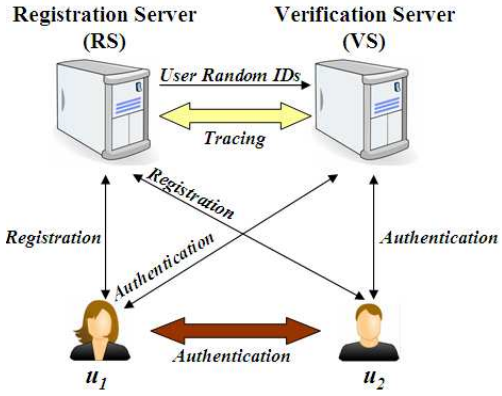

Figure 1: An Overview of the Data Flow

4 RAU: The Proposed Randomized Authentication Protocol

In this section, we present the proposed privacy-preserving authentication protocol. We first discuss the system setup and give an overview of the protocol. Then we elaborate on the details in each phase of the authentication.

4.1 An Overview of the Protocol

The proposed authentication system consists of two kinds of entities: authentication servers and mobile users. In the basic version, the system has two authentication servers, namely Registration Server (RS), and Verification Server (VS). The two servers collaborate with each other to conduct authentication for vehicle users, and hence none of them is able to track the user alone. In addition, we assume that vehicles can communicate with remote authentication servers via the Internet, following the assumptions adopted in previous works [29]. When designing the protocol, we aim to achieve the following security requirements:

- Authenticating message senders: We should verify that the message sender is a legitimate user in the network.
- Preserving user anonymity: The real identity of the message senders should not be known by peer users. Further, entities (e.g., peer users or an authentication server) should not be able to track a user's behavior in that they should not be able to link multiple messages to the same sender.
- Providing traceability: If necessary, the two authentication servers will be able to collaboratively retrieve a user's real identity.

The proposed authentication protocol has three main phases: (1) user registration, (2) user authentication, and (3) identity tracing. Figure 1 illustrates an overview of the data flow in the system. At the beginning, users register at the RS server. The RS server shares part of the information of users' pseudo identities with the VS server. Whenever users want to communicate with others, they can randomly generate pseudo identities which can be verified by the VS server. If there is any dispute, the two servers will conduct a tracing protocol to figure out the real identity of the suspect vehicle. The detailed steps in each phase will be presented in the following subsections. For clarity, Table 1 lists the frequently used notions in this paper.

4.2 User Registration

To begin with, the RS server generates its own public-private key pair using the Paillier encryption scheme, and the public key is known by any user who logs onto an ad-hoc network. Users will always communicate with the servers through a secure channel. Specifically, a session key between a user u and a server can

Table 1: List of Notations

Notation	Meaning
RAU	Randomized Authentication
RS	Registration Server
VS	Verification Server
E(x,r)	Encrypt x using RS' public key
D(y)	Decrypt the ciphertext y
ID_u	Real identity of user u
RID_u	Randomized identity of user u
τ_u	Randomization interval of user u
γ_u	Randomization seed of user u
r_u^i	Random number generated at i^{th} round

be generated using any well-known method, e.g., Diffie-Hellman key exchange protocol. The remaining communication between the server and the user will be encrypted using the session key only known to them. Similarly, the communication between the two authentication servers is also via a secure channel.

User registration is an on-going process. That is, a new user can join the system at any moment. To register, a user u sends the real identity (ID_u) such as driver license number[1] to the registration server (RS) via the secure channel. The RS server verifies u's identity by resorting to a third authority such as DMV (Department of Motor Vehicles) of the government. Once the identity of u is verified, the RS server computes an initial randomized authentication ID (RID_u^0) for user u as follows:

$$\mathrm{RID}_u^0 = E(\mathrm{ID}_u, r_u^0) \tag{3}$$

where $E(\mathrm{ID}_u, r_u^0)$ is a Paillier encryption of the identity of u with a random number r_u^0 using the RS' public key.

Then, the RS server sends RID_u^0 to both user u and the verification server (VS). Since RID_u^0 is encrypted using the RS server's public key, only the RS server is able to decrypt it and reveal the real identity of the user. The actual identity of the user is always kept secret from the verification server during the lifetime that the user. After user u is registered, both the RS and VS servers store the user's initial randomized authentication ID RID_u^0 in their local databases DB_{rs} and DB_{vs} respectively. The plain texts of the real identities are discarded by the RS server to prevent attackers from hacking the system and stealing the sensitive information. The registration protocol is illustrated in Figure 2. Note that all messages are encrypted using the corresponding session keys between the communicating parties. For clarity, we only include the content of the messages in the figure.

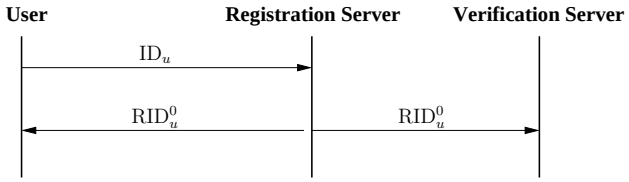

Figure 2: User Registration

4.3 User Authentication

Without loss of generality, we present the one-way authentication protocol for user u_2 (or a service provider) to verify if u_1 is a legitimate user. To achieve mutual authentication, the process can be executed again

[1]Here we use driver license number for illustration only. In practice, we can use more complex information to verify a user's identity to prevent an adversary from guessing.

with u_1 and u_2 by switching their roles. The authentication protocol consists of three phases: (i) identity validation, (i) ownership validation, and (iii) generation of randomized authentication ID.

4.3.1 Identity Validation

Suppose u_1 uses his or her randomized authentication ID $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ to initiate the identify validation process with user u_2. First, u_1 executes the Diffie-Hellman key exchange protocol with u_2 to mutually generate a random number $k_{u_1u_2}$. The protocol guarantees that the probability of other two users obtaining the same random number $k_{u_1u_2}$ is close to zero as long as one of the users follows the protocol. In other words, $k_{u_1u_2}$ is unique for each pair of user each time they execute the protocol. The use of this random number is to prevent the man-in-the-middle attack (discussed in Section 5).

Before sending $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ to user u_2, u_1 will first register a pending authentication request at the VS server by sending the message: $p_{u_1} = [\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}, k_{u_1u_2}]$. The VS server will searches its database to look for $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$. If $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ exists, the VS server will record this pending request. If not, the VS server will deny the authentication request. Upon receiving the acknowledgment of successful registration of authentication request from the VS server, u_1 sends $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ to u_2.

In addition, u_1 can also start concurrent authentication sessions with other users at this moment. For example, suppose u_1 wants to contact with three other users (e.g., u_3, u_4 and u_5), u_1 can ask RS to generate three new randomized authentication IDs (see Section 4.3.3 for technical details), and start three additional sessions with VS as stated above, and continues with the following steps with each user. Here we use u_2 to illustrate these steps. For u_2 to verify the received $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$, user u_2 forwards this randomized ID together with the random number $k_{u_1u_2}$ to the VS server. If the VS server finds a pending authentication request that matches the message sent by u_2, the VS server will inform u_2 that this is a valid ID. Otherwise, the VS server will inform u_2 that authentication fails.

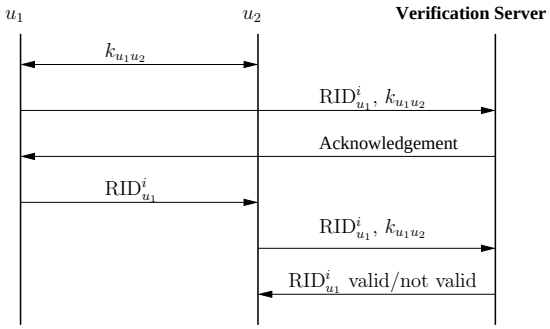

Figure 3: Identity Validation

4.3.2 Ownership Validation

Once user u_2 confirms the validity of $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$, user u_2 may need to further verify whether user u_1 is the real owner of this ID. If $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ is currently stored at VS (i.e., has yet to be used by u_1), this step prevents an adversary to use $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ to authenticate his or herself with other users. For instance, an attacker may hack into the server's system or the user's device to obtain a copy of the current randomized IDs of some users, not the private key which is assumed to be securely stored. Unless the malicious user continuously monitors or fully controls the server's system which is usually very difficult not to be detected by the server, he/she would not be able to impersonate other users using the obtained one-time randomized IDs because he/she cannot successfully pass the following ownership validation step. Note that this step is optional and not necessary if the user and the servers' systems are reasonably secure.

First, user u_2 selects two random values c and r, and sends the following value v_1 to u_1.

$$v_1 = \left(\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}\right)^c * E(0, r) \tag{4}$$

where c is a challenging value for u_1 to discover, and r can be any random number just for performing the encryption of 0. The purpose of multiplying with $E(0, r)$ is to randomize $\left(\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}\right)^c$, so that it is computationally infeasible for an adversary to compute the discrete log of $\left(\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}\right)^c$ to obtain c. Only if user u_1 is the real identity owner, u_1 will be able to compute the encrypted value of the challenging value c. Specifically, u_1 first encrypts the multiplicative inverse of his or her real identity $\mathrm{ID}^{-1}_{u_1}$. Then, u_1 computes a value v_2 by $v_1^{\mathrm{ID}^{-1}_{u_1}}$. According to the properties of homomorphic encryption, value v_2 is equal to the encrypted value of c as deduced as follows:

$$\begin{aligned} v_2 &= v_1^{\mathrm{ID}^{-1}_{u_1}} \\ &= \left(\left(\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}\right)^c * E(0, r)\right)^{\mathrm{ID}^{-1}_{u_1}} \\ &= E(c * \mathrm{ID}_{u_1} * \mathrm{ID}^{-1}_{u_1}, r') \\ &= E(c, r') \end{aligned}$$

Then, user u_1 sends v_2 to VS who asks the RS server to decrypt $v_2 = E(c, r')$ and obtain an decrypted value $D(E(c, r'))$. This $E(c, r')$ does not contain any identity information about user u_1, and hence the RS server does not know whose identity that u_2 is trying to verify. Upon receiving the decrypted value from RS, VS sends the value to u_2. Then u_2 checks if $D(E(c, r'))$ equals to c. If yes, user u_2 knows that u_1 is the real owner of the identity and informs the VS the authentication succeeded. Figure 4.3.2 depicts the main messages exchanged during this validation phase. For the above scheme to work, $\mathrm{ID}_{\mathrm{u}_1}$ needs to have a multiplicative inverse in $\mathbb{Z}_N$. Since $N = pq$, and p and q are very large prime numbers, this requirement can be easily satisfied.

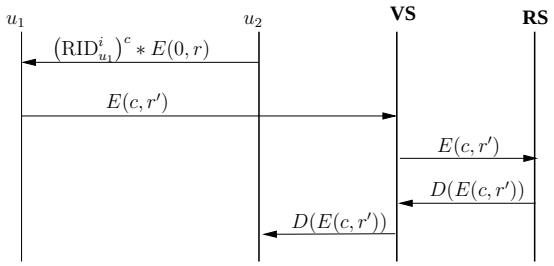

Figure 4: Ownership Validation

4.3.3 Generation of Randomized Authentication ID

In our system, each randomized authentication ID is only used once so that users' behavior will not be tracked by any party. When user u_1 after communicates with user u_2, user u_1 needs to acquire a new randomized authentication identity $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ from the RS server. After u_1 informs RS its intention to obtain a new randomized ID, the RS server can produce one based on the last ID associated with u_1 according to Equation 5.

$$\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1} = \mathrm{RID}^{\mathrm{i}-1}_{\mathrm{u}_1} * E(0, r^i_{u_1}) \tag{5}$$

The $r^i_{u_1}$ value is randomly generated for this i^{th} request from u_i. Based on the addition property of the homomorphic encryption (Equation 1), the new randomized ID is again the encryption of the real identity which can be deduced as:

$$\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1} = E(\mathrm{ID}_{\mathrm{u}_1}, r^{i-1}_{u_1}) * E(0, r^i_{u_1}) = E(\mathrm{ID}_{\mathrm{u}_1} + 0, r')$$

It is worth mentioning that by leveraging this addition property, the generation of new randomized ID using the above Equation is more efficient and secure than directly encrypting the real identity again since there is no need for the RS server to keep the real identity after the registration phase. This is why the RS server should delete a user's real identity for added security protection.

The RS server sends the newly generated randomized ID to both u_1 and VS. When the VS server received the new copy of randomized IDs, it would not be able to link each new ID to its previous version. It is important to note that the main purpose of generating a new set of randomized authentication IDs is to prevent VS from tracking the number of times a user has been authenticated. Through a set of randomized IDs, it is not clear if VS can discover anything related an actual user. Therefore, except for the initial user registration, periodically generating randomized authentication IDs may not be necessary. We provide such an option in case the users are exceedingly concerned about their privacy.

4.4 Identity Tracing

In some applications, disputes may occur due to various reasons. Sometimes a third-party law enforcement authority may want to know immediately the real identity of a suspect user who is undergoing an authentication. Sometimes there may be a need to discover the authentication history of a suspect user. Thus, we propose both real-time identity tracing and historical identity tracing.

The real-time identity tracing is easy to achieve. The law enforcement authority submits the tracing request that contains the suspect user's randomized authentication ID to either the VS server or the RS server. If the request is received by the VS server. The VS server will forward the suspect user's randomized ID to the RS server. Upon receiving the suspect user's randomized ID, the RS server uses its private key to decrypt the randomized ID and reports the real identity to the law enforcement authority.

In terms of historical identity tracing, the law enforcement authority captured one randomized ID of the suspect user and wants to know the authentication history of the user to figure out the user's behavior in the network. The law enforcement authority sends the randomized ID of the suspect user to both RS and VS server. The RS server maintains a list of authentication history of all users. For example, each user has a list of randomized authentication IDs that have been or are planning to be used. The VS servers maintains all valid authentication IDs and their targeted service providers or peer users like u_2 in our example.

First, the RS server find a match in a user's list. If there is a match, the list of randomized IDs will be provided to the law enforcement authority who will subsequently send these IDs to VS. The VS will return the authority the service providers who have provided services to the user with these randomized ID. Based on the location of the service providers, the authority may learn where the suspect has been before. To provide this kind of historical tracing, the only thing needs to be changed is that the RS and VS servers need more memory space to store previously used randomized IDs. In addition, when the VS server performs identity validation, it needs to make sure, old IDs cannot be used again. These modifications can be easily incorporated into our current scheme.

Finally, we would like to mention that the identity revocation is very efficient in our system. Once a suspect user is confirmed to be malicious, the RS and the VS server just need to remove this user's randomization ID from their database. Any subsequent authentication request for this malicious user will fail as no matching record will be found by the server any more.

5 Threat Models and Analysis

To analyze if a protocol is secure, first we need to be clear the threat or adversary models considered in our problem domain. Like all existing work discussed in Section 2, we assume the authentication servers: RS and VS are semi-honest. That is both RS and VS follow the prescribed procedures of the proposed protocol. This is a legitimate assumption if RS and VS are well-known IT companies. On the other hand, the mobile users or service providers can be malicious. That is they can do whatever they can to discover the real identity of a user.

In this section, we analyze the properties of our proposed authentication protocol and discuss its robustness against various types of attacks according to the above assumptions regarding the adversarial behaviors. The proposed authentication protocols have the following three properties: (i) unforgeability, (ii) full privacy preservation, and (iii) traceability.

5.1 Unforgeability

Our authentication protocol guarantees that no one can use the identity that does not belong to him/her. Under the assumptions that the private key is kept securely at the RS server side, the only option left for the attacker to impersonate legitimate users is to exploit their randomized authentication IDs. There are several possible ways for an attacker to obtain a randomized authentication ID of a user. However, we show in the following that the attacker would not be able to use this ID as its own for authentication purpose.

An attacker can obtain another user's valid authentication ID during authentication. However, the attacker cannot directly use the received authentication ID again since each ID is allowed to be used only once and is discarded after the use by the VS server. If the attacker tries to re-randomize the received ID using a new random number, the resulting ID will not match any valid authentication ID stored in the VS server. This is because the attacker does not know the randomization seed used by the real owner of the ID, and hence the attacker would not be able to generate the same series of randomized IDs that match the real ones.

Previous discussion is focused on the used randomized IDs. We now discuss the case when the attacker steal the new randomized IDs from the user, the VS or RS server. Since these IDs have not been used by the real owner, the attacker will be able to go through the user authentication phase, but will be caught at the ownership validation phase. This is because the attacker does not know the real identity of the ID owner, and hence the attacker cannot discover the random number included in the challenging question (as discussed in Section 4.3.2).

Alternatively, an attacker may try to perform the man-in-the-middle attack. As illustrated in Figure 5.1, the attacker u_m attempts to forward legitimate user u_1's authentication ID to another legitimate user u_2, and vice versa. The attacker aims to prove to u_1 that he is user u_2, and prove to u_2 that he is user u_1. However, such attack will not succeed because our protocol verifies a mutually agreed random number between each pair of users and this random number is unique for each pair of users at each round of generation. Recall that at the beginning of the user authentication (Section 4.3.1), user u_1 and the attacker generates a mutually agreed random number ($k_{u_1 u_m}$). User u_1 registered this random number at the VS server along with its authentication ID ($\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$). For anyone who wish to verify the validity of ($\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$), he/she needs to provide the random number ($k_{u_1 u_2}$) to the VS server to prove that he/she is the person who u_1 is currently communicating with. Therefore, even if the attacker tries to present the obtained ($\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$) to u_2, the attacker would not be able to establish a mutually agreed random number with u_2 that is the same as ($k_{u_1 u_2}$), as long as u_2 does not collude with u_m (this is the general assumption under the man-in-the-middle attack). Consequently, if u_2 presents $\mathrm{RID}^{\mathrm{i}}_{\mathrm{u}_1}$ and a different random number say $k_{u_m u_2}$ to the VS server for verification, the VS server will easily discover the matching IDs but un-matching random numbers and conclude that there is a man-in-the-middle attack.

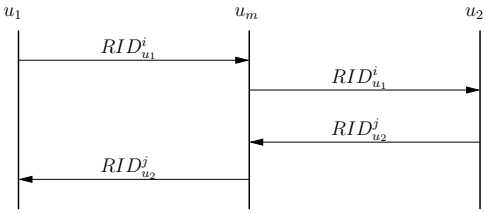

Figure 5: Man-in-the-middle Attack

5.2 Full Privacy Preservation

Our authentication protocol provides full privacy preservation in that it guarantees both server-wise and peer-wise privacy for the users in terms of both anonymity and unlinkability. In particular, a user can always self-generate a new randomized authentication ID when establishing a new communication session. Thus, peer users would not know the real identity of others, nor be able to link different communication sessions to the same user.

As for the VS server, it does not have the key to decrypt the randomized IDs stored in its database, and hence it does not know the real identity of the user who submits authentication request. As for the RS server, it does not handle any authentication request that contains randomized IDs during the authentication phase, and hence the RS server does not know who is sending the authentication request and cannot track the users.

5.3 Traceability

Traceability refers to the ability to reveal the user's real identity requested by the law authorities. This is a seemingly conflicting requirement with respect to the privacy preservation goal of our system. We achieve this by proposing the collaborative identity tracing protocol as presented in Section 4.4. The identity tracing protocol is capable of revealing a suspect user's real identity and his/her whole authentication history to the law authorities without violating the privacy of other legitimate users.

6 Experimental Study

Please note that our protocol does not require the users be equipped with high performance computing equipment. The following hardware specification is used to simulate the servers, but not the hardware on the vehicles. We implemented the RAU authentication protocol in C language with GMP library, and run the tests on a PC with Intel Xeon CPU X5675 @3.07GHZ x6 and 11.7GB memory. We evaluate the efficiency of the total authentication process in terms of computational cost. We did not include the transmission and propagation delay since they are dependent on specific network configuration.

6.1 User Registration

The main computational cost involved in user registration is the generation of the initial randomized ID for the new user. Each randomized ID is 2048 bits. From the experiments, we observe that the randomized ID generation time is less than 3ms for each user.

6.2 User Authentication

Recall that the user authentication protocol consists of three phases: identity validation, ownership validation and randomized ID generation. Please note that not every phase is required for each user authentication. As discussed in Section 4.3.2, the ownership validation phase is optional. Thus, the efficiency of the identity validation phase is important. As reported later in this section, all three phase combined take about 7ms. In addition, the average latency for a 4G network (e.g., Verizon) is 30ms. Therefore, the total cost is well below the 100ms requirement stated in [20] (for VANET). Identity tracing is very efficient, and it only requires several encryption operations.

As mentioned in Section 1, the anonymous credential (AC) protocol [3] uses zero-knowledge proof to achieve privacy-preserving user authentication. Being the best among the exiting solutions, that scheme achieves most criteria for anonymous authentication defined in this paper. Here we compare its efficiency with our protocol. In order to have the same security guarantee as our proposed RAU protocol, we need to include the run time for both credential issuing and verification phases. We implement the AC protocol under the same computing environment as RAU, and the run time for each protocol is reported in Table 2. According to the table, 7ms include all three phases of RAU. RAU- represents the situation where the

Protocol	Run time
RAU	7ms
RAU-	3ms
AC	40ms

Table 2: RAU vs. AC

ownership validation phase is omitted. As we can see, the proposed protocol is considerably more efficient than the AC protocol.

7 Conclusion and Future Work

In this paper, we present a randomized authentication protocol for mobile users in both MANET and VANET. The protocol leverages the properties of an additive homomorphic probabilistic public key encryption system. The proposed protocol overcomes shortcomings in other existing works, and achieves a set of desired properties including unforgeability, full privacy preservation, identity tracing and fault tolerance. In particular, it is more efficient and secure than [3]. In the future, we plan to study access control mechanism in mobile environments that can be integrated into the proposed anonymous authentication system.

References

[1] Randy Baden, Adam Bender, Neil Spring, Bobby Bhattacharjee, and Daniel Starin. Persona: an online social network with user-defined privacy. In *ACM SIGCOMM Computer Communication Review*, volume 39, pages 135–146. ACM, 2009.

[2] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux, and Antonio Lioy. Efficient and robust pseudonymous authentication in vanet. In *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, pages 19–28. ACM, 2007.

[3] J. Camenisch and E. V. Herreweghen. Design and Implementation of the idemix Anonymous Credential System. In *ACM Computer and Communications Security Conference*, pages 21–30, 2002.

[4] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *EUROCRYPT*, pages 93–118, 2001.

[5] Emanuele Cesena, Hans Löhr, Gianluca Ramunno, Ahmad-Reza Sadeghi, and Davide Vernizzi. Anonymous authentication with tls and daa. In *Trust and Trustworthy Computing*, pages 47–62. Springer, 2010.

[6] D. Chaum and E. V. Heijst. Group signatures. In *Advanced CryptologyEurocryptS*, pages 257–265, 1991.

[7] N. Chen, M. Gerla, D.Huang, and X. Hong. Secure, selective group broadcast in vehicular networks using dynamic attribute based encryption. In *Ad Hoc Networking Workshop (Med-Hoc-Net), 2010 The 9th IFIP Annual Mediterranean*, pages 1–8. IEEE, 2010.

[8] Chao-Chin Chou, David SL Wei, C-CJ Kuo, and Kshirasagar Naik. An efficient anonymous communication protocol for peer-to-peer applications over mobile ad-hoc networks. *Selected Areas in Communications, IEEE Journal*, 25(1):192–203, 2007.

[9] Tomasz Ciszkowski and Zbigniew Kotulski. Anap: Anonymous authentication protocol in mobile ad hoc networks. *arXiv preprint cs/0609016*, 2006.

[10] Whit Diffie and Martin Hellman. New directions in cryptography. IT-22(6):644–654, November 1976.

[11] Julien Freudiger, Maxim Raya, and Jean-Pierre Hubaux. Self-organized anonymous authentication in mobile ad hoc networks. In *Security and Privacy in Communication Networks*, pages 350–372. Springer, 2009.

[12] C. Gamage, B. Gras, B. Crispo, and A.S. Tanenbaum. An identity-based ring signature scheme with enhanced privacy. In *Securecomm and Workshops*, pages 1–5, 2006.

[13] G.Calandriello, P. Papadimitratos, J.P. Hubaux, and A. Lio. Efficient and robust pseudonymous authentication in vanet. In *Proc. of the 4th ACM international workshop on Vehicular ad hoc networks*, pages 19–28, 2007.

[14] M. Gerla and M. Gruteser. Vehicular networks: Applications, protocols, and testbeds. *In Emerging Wireless Technologies and the Future Mobile Internet*, pages 201–241, 2011.

[15] Shafi Goldwasser, Silvio Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pages 291–304, Providence, Rhode Island, U.S.A., May 6-8 1985.

[16] Y. Hao, C. Yu, C. Zhou, and W. Song. A distributed key management framework with cooperative message authentication in vanets. *Selected Areas in Communications, IEEE Journal on*, 29(3):616–629, 2011.

[17] RIFA-POUS Helena, Emmanouil A Panaousis, and Christos Politis. Recipients' anonymity in multihop ad-hoc networks. *IEICE TRANSACTIONS on Information and Systems*, 95(1):181–184, 2012.

[18] Xuedan Jia, Xiaopeng Yuan, Lixia Meng, and Liangmin Wang. Epas: Efficient privacy-preserving authentication scheme for vanets-based emergency communication. *Journal of Software*, 8(8):1914–1922, 2013.

[19] C. D. Jung, C. Sur, Y. Park, and K.-H. Rhee. A robust conditional privacy-preserving authentication protocol in vanet. *Social Informatics and Telecommunications Engineering*, 17:35–45, 2009.

[20] Georgios Karagiannis, Onur Altintas, Eylem Ekici, Geert Heijenk, Boangoat Jarupan, Kenneth Lin, and Timothy Weil. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. *IEEE Communications Surveys & Tutorials*, 13(4):584–616, 2011.

[21] Panayiotis Kotzanikolaou, Emmanouil Magkos, Nikolaos Petrakos, Christos Douligeris, and Vassilis Chrissikopoulos. Fair anonymous authentication for location based services. In *Data Privacy Management and Autonomous Spontaneous Security*, pages 1–14. Springer, 2013.

[22] X. Lin, X. Sun, P.-H. Ho, and X. Shen. Gsis: A secure and privacy-preserving protocol for vehicular communications. *IEEE Transactions on Vehicular Technology*, 56(6):3442 – 3456, 2007.

[23] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen. Ecpp:efficient conditional privacy preservation protocol for secure vehicular communications. In *Proc. of IEEE Conference on Computer Communications*, pages 1229 – 1237, 2008.

[24] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe. Parknet: drive-by sensing of road-side parking statistics. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys 2010)*, pages 123–136, 2010.

[25] P. Paillier. Public key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology - Eurocrypt '99 Proceedings, LNCS 1592*, pages 223–238, Prague, Czech Republic, May 2-6 1999. Springer-Verlag.

[26] M. Raya and J. P. Hubaux. Securing vehicular ad hoc networks. In *Journal of Computer Security*, pages 39–68, 2007.

[27] Maxim Raya, Daniel Jungels, Panos Papadimitratos, Imad Aad, and Jean-Pierre Hubaux. Certificate revocation in vehicular networks. *Laboratory for computer Communications and Applications (LCA) School of Computer and Communication Sciences, EPFL, Switzerland*, 2006.

[28] Jian Ren and Lein Harn. An efficient threshold anonymous authentication scheme for privacy-preserving communications. 2013.

[29] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang. Adaptive privacy-preserving authentication in vehicular networks. In *Communications and Networking in China, 2006. ChinaCom'06. First International Conference on*, pages 1–8. IEEE, 2006.

[30] K.-A. Shim. Cpas: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks. In *IEEE Transaction on Vehicular Technology*, pages 1874–1883, 2012.

[31] A. Squicciarini, D. Lin, and A. Mancarella. Paim: Peer-based automobile identity management in vehicular ad-hoc network. In *Proc. of the IEEE Computer Software and Applications Conference (COMPSAC)*, 2011.

[32] A. Studer, E. Shi, F. Bai, and A. Perrig. Tacking together efficient authentication, revocation, and privacy in VANETs. In *Proceedings of the 6th Annual IEEE communications society conference on Sensor, Mesh and Ad Hoc Communications and Networks*, SECON'09, pages 484–492, Piscataway, NJ, USA, 2009. IEEE Press.

[33] J. Sun, C. Zhang, Y. Zhang, and Y. Fang. An identity-based security system for user privacy in vehicular ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(9):1227–1239, 2010.

[34] Xiaoting Sun. Anonymous, secure and efficient vehicular communications. Master's thesis, The University of Waterloo, Waterloo, Ontario, Canada, 2007.

[35] Yipin Sun, Rongxing Lu, Xiaodong Lin, Xuemin Shen, and Jinshu Su. An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications. *Vehicular Technology, IEEE Transactions on*, 59(7):3589–3603, 2010.

[36] Z. Tan. A privacy-preserving mutual authentication protocol for vehicle ad hoc net. *Journal of Convergence Information Technology*, 5(7), 2010.

[37] Z. Tan. A lightweight conditional privacy-preserving authentication and access control scheme for pervasive computing environments. *Journal of Network and Computer Applications*, 2012.

[38] Jia-Lun Tsai, Nai-Wei Lo, and Tzong-Chen Wu. Secure anonymous authentication protocol with unlinkability for mobile wireless environment. In *Anti-Counterfeiting, Security and Identification (ASID), 2012*, pages 1–5. IEEE, 2012.

[39] W. Viriyasitavat, F. Bai, and O. K Tonguz. Toward End-to-end Control in VANETs. In *IEEEVehicular Networking Conference (VNC)*, pages 78–85, 2011.

[40] L.Y. Yeh, Y.C. Chen, and J.L. Huang. Paacp: A portable privacy-preserving authentication and access control protocol in vehicular ad hoc networks. *Computer Communications*, 34(3):447–456, 2011.

[41] C. Zhang, P.-H. Ho, and J. Tapolcai. On batch verification with group testing for vehicular communications. *Wireless Networks*, 17(8):1851–1865, 2011.

[42] C. Zhang, R. Lu, X. Lin, P.H. Ho, and X. Shen. An efficient identity-based batch verification scheme for vehicular sensor networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 246–250. IEEE, 2008.

[43] J. Zhang, Y. Cui, and Z. Chen. Spa: Self-certified pkc-based privacy-preserving authentication protocol for vehicular ad hoc networks.